

- Over **14+ years** of professional experience as software professional industry comprising of delivery management, design, development, deployment and cloud implementation with thorough knowledge in AWS (Amazon Web Services), Hybrid Cloud, VMware virtualization/ Windows, and related technologies.
- Extensive experience in Implementing Organization Cloud Engineer strategy based on Amazon Web Services.
- AWS (Amazon Web Services): EC2, VPC, IAM, IAM, S3, Cloud Front, Cloud Watch, Cloud formation, Glacier, RDS Config, Route 53, SNS, SQS, Elastic Cache.
- Utilized Cloud Watch to monitor resources such as EC2, CPU memory, Amazon RDS DB Services, EBS volumes; to set alarm for Notification or Automated actions, and to monitor logs for better understanding and operation of the system.
- Ability to Maintain High availability infrastructure in AWS, and to process Requests like Creation of Servers.
- Experienced with installation of AWS CLI to control various AWS services through SHELL/BASH scripting.
- Possess working knowledge with Python to automate software configuration.
- Experienced in creating multiple VPC's and public, private subnets as per requirement and distributed them as groups into various availability zones of the VPC.
- Created and configured elastic load balancers and auto scaling groups to distribute the traffic and to have a cost efficient, fault tolerant and highly available environment.
- Created S3 buckets in the AWS environment to store files, sometimes which are required to serve static content for a web application.
- Configured S3 buckets with various life cycle policies to archive the infrequently accessed data to storage classes based on requirement.
- Used IAM for creating roles, users, and groups to provide additional security to AWS account and its resources.
- Experienced in creating RDS instances to serve data through servers for responding to requests.
- Created snapshots and also images to store launch configurations of the EC2 instances. Good knowledge in Continuous Integration (CI) and Continuous Deployment(CD) methodologies
- Provided support for java applications by collaborating with java development team using the agile methodology
- Managed and automated all aspects of our AWS infrastructure (compute, storage, network, permissions, cost) using configuration management tools like Ansible, Cloud Formation and shell scripts
- Wrote Ansible playbooks for automating the process of provisioning custom Docker containers using AWS EC2 machines as hosts
- Assist in designing, automating, implementing and sustainment of Amazon machine images (AMI) across the AWS Cloud environment
- Experience with AWS API Gateway and Rest APIs.
- Experience with NoSQL data-stores like HBase, Cassandra
- Created and modified Cloud Formation templates to create/upgrade EC2 instances to support specific needs
- Used Vagrant to create compute layer which in turn creates Virtual Machines depending on scalability
- Used Dockerfiles to create Docker images that will contain preinstalled software and micro-services
- Experience in working on several Docker components like Docker engine, Hub, Machine, Compose and Docker registry
- Configured Jenkins to automate the build deployments based on the changes captured in Version Control,
- Expertise in Ansible Playbook, YAML for Maintaining Roles, Inventory Files and Groups Variables
- Experience in branching, tagging and maintaining the version control and source code management tools like GIT, SVN (subversion) on Linux.

**PROFESSIONAL EXPERIENCE:**

**Lead AWS DevOps**

**Goldman Sachs. NY**

**March 2022 to June 2024**

**Responsibilities:**

- Experience in Server infrastructure development on AWS Cloud. extensive usage of Virtual Private Cloud (VPC), Cloud Formation, Cloud Front, Elastic Compute Cloud EC2, RDS, S3, Route53, SNS, SQS, and Cloud Trail.
- Created the ACLs. Subnets. Route Tables. Internet Gateway and NAT Gateway.
- Experience in configuration and administration of Load Balancers, Network and Auto scaling for sub domains in AWS VPC.
- Responsible to Author ISON files to create stacks on AWS Cloud Formation
- Provide access to different user base using AWS IAM
- Trained 9 people in DevOps and OpsSuite team to the extent of being able to do the POC sub tasks.
- Extensive usage of different AWS Web Services like, EC2, S3, SWF, SNS, SQS, Dynamo DB Etc.,
- Responsible to migrate code from SVN to GIT
- Designed the network architecture for the POC.
- Experience using and creating custom scripts and applications using the AWS SDK and/or CLI Tools.
- Demonstrated the capacity to deploy complex infrastructure using CloudFormation.
- Automated the code base migration from SVN to GIT using BASH
- Responsible to migrate the build scripts from Maven to Gradle
- Responsible to migrate jobs on legacy/on premise Jenkins to AWS Jenkins
- Responsible to create git repositories on stash. Responsible for setting up End to End continuous integration pipeline for different Java Web Services manage permissions & enable Hooks
- Worked on restructuring the entire Gem fire and OpsSuite applications clusters to work with CloudFormation.
- Responsible for promoting different Java Web Services via Jenkins to push artifacts to Nexus
- Supporting team with all kinds of IBM Build forge issues.
- Deploying Java Web Services to Dev and Integration Test Environments using Run Deck.

**AWS Lead Devops**

**American Express. NY**

**January 2021 to February 2022**

**Responsibilities:**

- Experience with software release management (GIT) and agile/scrum methodologies
- Solid back end experience with Amazon Web Services (EC2, S3, ELB). deployment systems (chef, puppet)
- Experience with build & configuration management tools (Chef, Terraform, Puppet)
- Solid familiarity with building and maintaining continuous deployment environments
- Deployment and management through AWS cloud formation on EC2 (Elastic compute cloud) instances and also maintaining amazon S3 storage.
- Optimized volumes and EC2 instances & Created multi AZ VPC instances & Used IAM to create new accounts, roles and groups.
- Designed and implemented Amazon Web Services architecture, API implementation EC2 Server setup and deployment, build, maintenance, and configuration of various AWS resources like, EC2, EBS, Elastic Load Balancers, S3, VPC, and SecurityGroups that are utilized for different environments like dev, testing, and staging/prod.

- Experience in deploying custom chef cookbooks and puppet modules, to configure the machines indifferent environments with appropriate package/services and versions.
- Configured chef to manage AWS environments and automate the build process for core AMIs used by all application deployments including AutoScaling, and incorporating chef on the JSON CloudFormation scripts for the configuration deployment onto the nodes.
- Wrote and maintained extensive documentation & created cookbooks using Ruby framework for configuring the infrastructure and application servers.
- Monitoring of network services (SMTP, POP3, HTTP, NNTP, ICMP, SNMP, FTP, SSH) using Nagios.
- Installed and configured Nexus to manage the artifacts in different Repositories and handling dependency management using nexus private repository.
- Working on Video platform (JW Player) and monetizing the video Content.
- Utilized CloudWatch to monitor resources such as EC2, CPU memory, Amazon RDS DB services, DynamoDB tables, EBS volumes; to set alarms for notification or automated actions; and to monitor logs for a better understanding and operation of the system.

#### **AWS DevOps Engineer**

**Morgan Stanley. NY**

**August 2020 to December 2021**

#### **Responsibilities:**

- Deployment and management through AWS cloud formation on EC2 (Elastic compute cloud) instances and also maintaining amazon S3 storage.
- Optimized volumes and EC2 instances& Created multi AZVPC instances& Used IAM to create new accounts, roles and groups.
- Experience in deploying custom chef cookbooks and puppet modules, to configure the machines indifferent environments with appropriate package/services and versions.
- Configured S3 to host static web content, versioning and lifecycle policies to and backup files and archive files in Glacier.
- Proficient in using file extensions such as. DFXP & VTT for open and play video in Timed text format and media players such as JW player and Bitmovin.
- Configured chef to manage AWS environments and automate the build process for core AMIs used by all application deployments including AutoScaling, and incorporating chef on the JSON CloudFormation scripts for the configuration deployment onto the nodes.
- Wrote and maintained extensive documentation & created cookbooks using Ruby framework for configuring the infrastructure and application servers.
- Monitoring of network services (SMTP, POP3, HTTP, NNTP, ICMP, SNMP, FTP, SSH) using Nagios.

#### **AWS/DevOps Engineer**

**MetLife. NY**

**October 2018 to July 2020**

#### **Responsibilities:**

- Implemented and automated continuous build processes that build and deploy, identifies build errors and notifies responsible teams.
- Launched Amazon EC2 Cloud Instances, created S3 buckets, customized JSON template and Glacier, and administered and supported Git source code tool.
- Install, configure, and troubleshoot Red Hat Enterprise Linux. Well-versed in AWS Elastic Beanstalk for application deployments and operated on AWS lambda with Amazon kinesis.
- Migrated all the services to AWS.
- Eloqua Integration, Vast usage of Views, template design, JW player, Lightbox Integration
- Developed and utilized CloudWatch alerts in auto scaling launch configurations implemented and supported monitoring and alerting of production and corporate servers/storage via AWS CloudWatch.

- Automated log rotation of multiple logs from web serves by writing scripts in Python.
- Supported design and deployment of applications by employing AWS stack (EC2, Route 53, S3, RDS, Dynamo DB, SNS, SQS, IAM) with a focus on high availability, fault tolerance, and auto scaling in AWS CloudFormation.
- Managed AWS services EC2 instances, S3 EFS, VPC, Route 53
- Created Cloud watch agents with Lambda function using Python scripts.
- Implemented snapshots using CloudWatch.
- Built Cloud formation templates and created stacks using the templates for automations including deploying and events like snapshots, deploying Load balancers that are needed for AEM.
- Built dynamic routing/load balancing capability enabling large application scaling, used ingress rules and Ingress Controllers.
- Managed Docker DDC, Docker UCP
- Managed code commit and code deploy in AWS developer tool services.
- Forwarded log and taking backups to S3 bucket.
- Mounted EFS on multiple AWS instances.
- Managed and created different security groups under private VPC, domain names on Route 53.
- Experienced in security identity and compliance on AWS like IAM.
- Created required components to enable a private, internal only cloud infrastructure.
- Developed, maintained and enhanced pre-and post-build scripts (Shell and Python)
- Designed and documented CI/CD tools configuration management.
- Automated various infrastructure activities like Continuous Deployment, Application Server setup, Stack monitoring using Ansible playbooks and has Integrated Ansible with Jenkins.
- Handled Jenkins related activities such as creating new users with the right privileges, configuring the new builds, add/delete Jenkins slaves and resolving any Jenkins related issue.
- Implemented the setup for Master slave architecture to improve the Performance of Jenkins.
- Developed a continuous deployment pipeline using Jenkins, Ruby/shell scripts generated procedures.
- Reduced build + deployment times by designing and implementing Docker workflow.
- Responsible for all aspects of the Software Configuration Management (SCM) process including Code Compilation, Packaging, Deployment, Release Methodology and Application Configurations.
- Designed distributed private cloud system solution using Kubernetes (Docker) on CoreOS.
- Developed Puppet scripts for an automated server provisioning and Docker images for isolation, reducing the time between provisioning and deployment.
- Used confluence for documentation.
- Extensively worked on Linux, Redhat environment.

## **AWS Devops**

**Wells Fargo. San Francisco, CA**

**June 2016 to September 2018**

### **Responsibilities:**

- Worked with IAM service creating new IAM users & groups, defining roles and policies and Identity providers.
- Configure AWS IAM and Security Group in Public and Private Subnets in VPC.
- Involved in designing and deploying multitude applications utilizing almost all the AWS stack (Including EC2, Route53, S3, RDS, Dynamo DB, SNS, SQS, IAM) focusing on high-availability, fault tolerance, and auto-scaling in AWS Cloud Formation.
- Build servers using AWS, importing volumes, launching EC2, RDS, creating security groups, auto-scaling, load balancers (ELBs) in the defined virtual private connection.
- Created AWS Route53 to route traffic between different regions.
- Leveraged AWS cloud services such as EC2, auto-scaling, and VPC to build secure, highly scalable and flexible systems that handled expected and unexpected load bursts.
- Used AWS Beanstalk for deploying and scaling web applications and services developed with Java, PHP, Node.js, Python, Ruby, and Docker on familiar servers such as Apache, and IIS.

- Understanding of secure cloud configuration, (Cloud Trail, AWS Configuration), cloud-security Technologies (VPC, Security Groups, etc.) and cloud-permission systems (IAM).
- Created alarms and trigger points in Cloud Watch based on thresholds and monitored logs via metric filters.
- Utilized AWS cloud watch to monitor the performance environment instances for operational and performance metrics during load testing.
- Designed AWS Cloud Formation templates to create custom sized VPC, subnets, NAT to ensure successful deployment of Web applications and database templates.
- Implemented automated local user provisioning instances created in AWS cloud.
- Setup and build AWS infrastructure various resources, VPC EC2, S3, IAM, EBS, Security Group, Auto Scaling, and RDS in Cloud Formation JSON templates.
- Maintained the user accounts (IAM), RDS, Route 53, VPC, RDB, Dynamo DB, SES, SQS and SNS services in AWS cloud.
- Manage AWS EC2 instances utilizing Auto Scaling, Elastic Load Balancing and Glacier for our QA and UAT environments as well as infrastructure servers for GIT and Chef.
- Created customized AMIs based on already existing AWS EC2 instances by using create image functionality, hence using this snapshot for disaster recovery
- Created AWS Launch configurations based on customized AMI and use this launch configuration to configure auto-scaling groups and Implemented AWS solutions using EC2, S3, RDS, Dynamo DB, Route53, EBS, Elastic Load Balancer, Auto-scaling groups.

**AWS Devops**  
**Citigroup. NY**  
**October 2014 to May 2016**

#### **Responsibilities:**

- Worked within the Cloud integration processes.
- Worked on installation and configuration of Configuration management tool puppet 3.x, 4.x.
- Deployment and implementation of puppet in Redhat servers.
- Wrote Puppet Manifest files and Modules to deploy automated tasks to multiple servers.
- Implemented Git for version control of puppet modules and process documentation.
- Install and configure DHCP, web (Apache, IIS), mail (SMTP, IMAP, POP3), and file servers on Linux servers.
- Created and managed user accounts, security, Access Control Limits (ACL), permissions, disk space and process monitoring in Redhat Linux.
- Worked on AWS cloud, migrated application from physical servers to cloud.
- Hands on experience creating and managing instances in AWS cloud.
- Automated the deployments using Jenkins.
- Extensively worked on performance tuning of Applications and web servers.
- Monitored Disk & Memory Utilization on Linux 5.1, 6.2
- Installing and setting up Oracle10i and WebSphere on Linux for the development team.
- Worked on Linux kernel, memory upgrades and swaps area. Red hat Linux Kick starts Installation.
- System performance monitoring and troubleshooting in Linux Environments 5.1, 6.2.
- Installation, Configuration & Upgrade of Linux operating systems running oracle Database.
- Configuration and Administration of DNS, NIS, NIS+, NFS, Auto mount, DHCP, FTP, LDAP, Samba Send mail, Servers and clients. Security layers like SSL, SSH, HTTPS, SFTP, SSO
- Installation and configuration and maintenance of JBoss and Apache web server.
- Wrote various scripts on Bash, Shell and Ruby.

**AWS Devops**  
**Bank of America. Charlotte , NC**  
**August 2013 to September 2014**

### **Responsibilities:**

- Implemented and maintained monitoring and alerting of production and corporate servers such as EC2 and storage such as S3 buckets using AWS Cloud Watch.
- Experience in automated deployment of EC2 instances in data centers and availability zones.
- Used security groups to develop a logical firewall to ensure high security for control applications.
- Used Amazon IAM to grant fine access of AWS resources to users. Also managed roles and permissions of users to AWS account through IAM.
- Used Cloud Front to deliver content from AWS edge locations to users, allowing for further reduction of load on front-end servers.
- Set up Auto Scaling Groups based on memory and CPU to adapt to unforeseen spikes without having an outage or needing manual intervention.
- Set up Elastic Load Balancers for different applications to ensure high availability of applications.
- Used Amazon S3 to backup database instances periodically to save snapshots of data.
- Used Amazon Route53 to manage DNS zones and also give public DNS names to elastic load balancers ip's.
- Used Cloud Watch logs to move application logs to S3 and create alarms raised by applications.
- Configured an AWS Virtual Private Cloud (VPC) and Data Base subnet group for isolation of resources within AWS RDS.
- Used Amazon RDS Multi-AZ for automatic failover and high availability at the database tier for MYSQL workloads.
- Configured S3 Versioning and Life Cycle Policies to backup files and archive files in Glacier.
- To design high availability applications on AWS across Availability Zones and availability regions.
- Trained a group of employees about infrastructure management using AWS Management Console. .Developed and supported the Red Hat Enterprise Linux based infrastructure in the cloud environment.
- Hands on experience with Chef in AWS Cloud environment
- Designed continuous delivery tools/ configuration management tools like Puppet.
- Installed, configured and managed Puppet Master and Agent.
- Written CHEF cookbooks for various configurations to modularize and optimize end product configuration.
- Installed workstation, bootstrapped nodes, wrote recipes, cookbooks, roles and environments and uploaded them to chef server.

### **AWS Devops**

**JPMorgan Chase. New York, New York**

**March 2011 to July 2013**

### **Responsibilities:**

- Involved in designing and deploying multitude applications utilizing almost all of the AWS stack (Including EC2, Route53, S3, RDS, Dynamo DB, SNS, SQS, IAM) focusing on high-availability, fault tolerance, and auto scaling in AWS Cloud Formation.
- Working as an Aws Engineer/ Python Developer for supporting multiple projects doing automation for OS patching to the all Linux distributions and implemented monitoring tool in Python for getting agent logs from all the Linux virtual machines.
- Implemented python scripts by using stranded libraries for getting the all the agent logs (Inventory, Remote connections, Network usage and performance counters) from the different flavors of Linux.
- Configuration tool Installed and configured such as chef server / workstation and nodes via CLI tools to AWS nodes.
- Created private cloud using Kubernetes that supports DEV, TEST, and PROD environments.
- Automated VSTS (Visual Studio Team System) build and deployment to IAAS and PASS environment in Azure and developed build support utilities in PowerShell and Go language.
- Installed VMware ESXi 5.5 and 6, VSphere Server and VMware center Server in rack Servers.

- Leveraged AWS cloud services such as EC2, auto-scaling and VPC to build secure, highly scalable and flexible systems that handled expected and unexpected load bursts.
- Manage amazon redshift clusters such as launching the cluster and specifying the node type as well.
- Implemented automated local user provisioning instances created in AWS cloud.
- Setup and build AWS infrastructure various resources, VPC EC2, S3, IAM, EBS, Security Group, Auto Scaling, and RDS in Cloud Formation JSON templates.
- Provide highly durable and available data by using S3 data store, versioning, lifecycle policies, and create AMIs for mission critical production servers for backup.
- Maintained the user accounts (IAM), RDS, Route 53, VPC, RDB, Dynamo DB, SES, SQS and SNS services in AWS cloud.
- Good understanding of AWS CLI Command Reference and Multifactor Authentication on AWS.
- Worked on cloud watch to monitor the resources such as EC2 CPU memory, Amazon to design high availability applications on AWS across availability zones.
- Designed and implemented scalable, secure cloud architecture based on Amazon Web Services.
- Worked on Database like Cassandra, PostgreSQL, and MySQL.
- Deploy website updates utilizing Octopus Deploy, Windows Explorer and Remote Desktop Connection (RDP) and assign folder permissions in both staging and production environments.
- Built Continuous Integration environment Jenkins and Continuous delivery environment.
- Utilized Configuration Management Tool Chef & created Chef Cookbooks using recipes to automate system operations.
- Deployed applications on AWS by using Elastic Bean Stalk.
- Lead efforts in the design, implementation and maintenance of system and application monitoring using enterprise or 3rd party products (Splunk, Kibana/Elastic Search (ELK), AppDynamic, SCOM, AWC, etc.) to ensure highest system availability
- Worked on Agile Methodology and used JIRA for defect tracking.
- Extracted the data and administrated NoSQL databases RIAK and Dynamo DB
- Worked on NO SQL databases such as Mark logic, MongoDB and NEO4j.
- Tech Stack: AWS (API gateway, Lambda, Route 53, DynamoDB, SES, SQS, SNS)

## **AWS Devops**

**Berkshire Hathaway. Omaha, NE**

**July 2009 to February 2011**

### **Responsibilities:**

- Good experience in Amazon Web Services (AWS) environment and good knowledge of AWS services like Elastic Compute Cloud(EC2), Elastic Load-balancers, Elastic Container Service (Docker Containers), S3, Elastic Beanstalk, CloudFront, Elastic File system, RDS, DMS, VPC, Route53, CloudWatch, CloudTrail, CloudFormation, IAM.
- Initiating Amazon EC2 Cloud Instances using Amazon Web Services (Linux/ Ubuntu) and managing and configuring launched instances with respect to specific applications.
- Created detailed AWS Security Groups which behaved as virtual firewalls that controlled the traffic allowed reaching one or more AWS EC2 instances.
- Managing Multiple AWS instances, assigning the security groups, Elastic Load Balancer and AMIs.
- Auto scaling the instances to design cost effective, fault tolerant and highly reliable systems.
- Configuring of Virtual Private Cloud (VPC) with networking of subnets containing servers.
- Built S3 buckets and managed policies for S3 buckets and used S3 bucket and Glacier for storage and backup on AWS.
- Setting Cloud Watch alerts for instances and utilizing in Auto-scaling launch configurations for better performance.
- Developed and managed cloud VMs with AWS EC2 command line clients and management console.
- Included security groups, network ACLs, Internet Gateways, and Elastic IP's to ensure a safe area for organization in AWS public cloud.

- Created elastic load-balancers (ELB) for different applications to provide high performance of applications.
- Utilized Amazon Route53 to manage DNS zones and also assign public DNS names to elastic load balancers IP's.
- Used Amazon RDS which includes automatic failover and high availability at the database layer for MySQL workloads.
- Good experience with Dynamo DB and Redshift.
- Migrated test and use cases for applications to AWS cloud with public and private IP ranges to accelerate development productivity by decreasing test-run times.
- Used Amazon IAM service to grant permission to assigned users and manage their roles
- Utilized Cloud Front to transfer data from edge locations to users, thus enabling more decrement of load on front-end servers.